Component-Based Software Design

Hierarchical Real-Time Scheduling
lecture 3/4

Enrico Bini

March 24, 2015

Component-Based Software Design
Enrico Bini
Design problem
Model of VM

Outline

Component-Based Software Design
Enrico Bini
Design problem
Model of VM

The design problem over parallel VM

Premise: the terms multi-core, multiprocessor, parallel machine, etc. are all used interchangeably, as they all are machines capable to perform computation simultaneously.

What is the **best** *VM which can* **schedule** *a given* **real-time application***?*

Needed ingredients:

- A model of a parallel VM
- An application model
- Local scheduler of application over VM
- Notion of "best"
- Optimization method

Component-Based Software Design
Enrico Bini
Design problem
Model of VM

Outline

1. Design problem
2. Model of VM

Component-Based Software Design
Enrico Bini
Design problem
Model of VM

Main characteristic of multicore

Resource is available along *two dimensions*

- *horizontal* dimension over time
- *vertical* dimension over the number of processing units

In short:

- two machines of speed 0.5 and
- one machine of speed 1

are different.
Migration hypothesis:

1. migration/identical multiprocessor: the work can freely migrate over all CPUs and it takes the same amount of time
   - reasonable as first approximation
   - in reality migration has a non-uniform (cache hierarchies) cost

Component-Based Software Design
Enrico Bini
Design problem
Model of VM

VM schedule

A schedule is modeled by a set of n *VM schedule functions*

$$i \in \mathcal{N} \qquad s_i : \mathcal{T} \to 2^{\mathcal{M}}$$

such that

$$\forall t \in \mathcal{T},\ \forall i \neq j, \qquad s_i(t) \cap s_j(t) = \emptyset$$

that is no machine can be used simultaneously by two VMs.
Equivalently: $s_{i,k}(t)$ schedule of i-th VM over k-th CPU at t

$$i \in \mathcal{N}, \quad s_{i,k} : \mathcal{T} \to \{0,1\}$$

$$\forall t, k, \quad \sum_{i=1}^{n} s_{i,k}(t) \leq 1 \qquad \text{core } k \text{ at } t \text{ to at most 1 VM}$$

Legal schedules

Given a global scheduling algorithm $\mathcal{A}$, we denote by $\mathcal{L}_*$ the set of all possible legal VM scheduling functions, defined as

$$\mathcal{L}_* = \{(s_{i,k})_{i\in\mathcal{N},k\in\mathcal{M}} \;:\; \text{scheduling functions } (s_{i,k})_{i\in\mathcal{N},k\in\mathcal{M}} \text{ may be generated by } \mathcal{A}\},$$

while the set $\mathcal{L}_i$ of legal scheduling functions for i-th VM is defined as a projection of $\mathcal{L}_*$, that is

$$\mathcal{L}_i = \{(s_{i,k})_{k\in\mathcal{M}} \;:\; (s_{i,k})_{i\in\mathcal{N},k\in\mathcal{M}} \in \mathcal{L}_*\}.$$

Since we focus on one VM only, we drop the index i of the VM

$$\mathcal{L}_i \;\Rightarrow\; \mathcal{L}, \qquad s_{i,k}(t) \;\Rightarrow\; s_k(t)$$

k is the index of the assigned physical machine

Parallel supply upper bound function

Definition (*parallel supply upper (bound) function* of a VM)

Given a VM, we define its *parallel supply upper bound function* $\mathsf{psuf}_k(t)$ with maximum parallelism k as

$$\mathsf{psuf}_k(t) = \max_{t_0\in\mathcal{T},(s_1,\dots,s_m)\in\mathcal{L}} \int_{t_0}^{t_0+t} \min\left\{k, \sum_{\ell=1}^{m} s_\ell(x)\right\} dx.$$

More properties

Let us assume that at some instant t^* we know that

$$\mathsf{pslf}_k(t^*) \geq c^*, \tag{1}$$

From Lipschitz-continuity and (1), we find

$$\forall t\in[0,t^*] \quad \mathsf{pslf}_k(t) \geq \begin{cases} 0 & 0 \leq t \leq t^* - \frac{c^*}{k} \\ c^* + k(t-t^*) & t^* - \frac{c^*}{k} < t \leq t^* \end{cases}$$

Let q and r be quotient and rest of the Euclidean division of any $t \geq 0$ by t^*, that is

$$t = qt^* + r, \qquad q\in\mathbb{N},\; r\in[0,t^*).$$

Then, for any $t \geq 0$, the following lower bound holds

$$\mathsf{pslf}_k(t) = \mathsf{pslf}_k(\underbrace{t^* + \cdots + t^*}_{q \text{ times}} + r) \geq$$

$$\geq \underbrace{\mathsf{pslf}_k(t^*) + \cdots + \mathsf{pslf}_k(t^*)}_{q \text{ times}} + \mathsf{pslf}_k(r) = qc^* + \mathsf{pslf}_k(r).$$

Parallel supply lower bound function

Definition (*parallel supply lower (bound) function* of a VM)

Given a VM, we define its *parallel supply lower bound function* $\mathsf{pslf}_k(t)$ with maximum parallelism k as

$$\mathsf{pslf}_k(t) = \min_{t_0\in\mathcal{T},(s_1,\dots,s_m)\in\mathcal{L}} \int_{t_0}^{t_0+t} \min\left\{k, \sum_{\ell=1}^{m} s_\ell(x)\right\} dx.$$

[indentical hyp means $\sum s_\ell$]

"minimum amount of resource made available by the VM in every interval of length t, with parallelism at most k".

Notice that $\mathsf{pslf}_k(t)$ depends on

1. the length of the time interval t, and
2. the level of parallelism k,

to represent the bi-dimensional nature of the resource.

Properties

1. $\forall k,\; \mathsf{pslf}_k(0) = 0$ (no resource in empty interval),
2. $\forall k,\; \forall s \geq t \geq 0,\; \mathsf{pslf}_k(s) \geq \mathsf{pslf}_k(t)$ (non-decreasing)
3. $\forall k,\; \forall s,t \geq 0,\; \mathsf{pslf}_k(s+t) \geq \mathsf{pslf}_k(s) + \mathsf{pslf}_k(t)$ (super-additivity)
4. $\forall k,\; \forall s \geq t \geq 0,\; \mathsf{pslf}_k(s) - \mathsf{pslf}_k(t) \leq k(s-t)$ (Lipschitz continuous with factor k):
5. when needed, we extend $\forall k \leq 0,\; \mathsf{pslf}_k(t) = 0$
6. $\forall k,\; \forall t \geq 0,\; \mathsf{pslf}_k(t) \leq \mathsf{pslf}_{k+1}(t)$ (non-decr with k)
7. (def): $\bar{m} = \min\{k : \forall t,\; \mathsf{pslf}_k(t) = \mathsf{pslf}_{k+1}(t)\}$ is *max parallelism* of the VM;
8. $\forall k, \forall t \geq 0,\; \mathsf{pslf}_k(t) - \mathsf{pslf}_{k-1}(t) \geq \mathsf{pslf}_{k+1}(t) - \mathsf{pslf}_k(t)$ (concavity) **[explanation]**
   implies $\forall k, \forall t \geq 0,\; \frac{\mathsf{pslf}_k(t)}{k} \geq \frac{\mathsf{pslf}_{k+1}(t)}{k+1}$ **[by induction]**
9. $\mathsf{pslf}_{\bar{m}}(b-a) \leq \int_a^b \sum_{k=1}^{m} s_k(t)\,dt \leq \mathsf{psuf}_{\bar{m}}(b-a)$

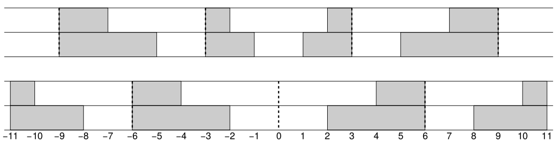
Examples

1. $\mathsf{pslf}_k(t)$ of a VM with $\bar{m}$ dedicated processors?
2. $\mathsf{pslf}_k(t)$ of a VM which provides Q budget, every period P, with parallelism at most $\bar{m}$?
3. $\mathsf{pslf}_k(t)$ of a VM which provides $Q_1 = 4, Q_2 = 2$ budgets, on two CPUs, every period $P = 6$?

Examples

1. $\mathsf{pslf}_k(t)$ of a VM with $\bar{m}$ dedicated processors?
2. $\mathsf{pslf}_k(t)$ of a VM which provides Q budget, every period P, with parallelism at most $\bar{m}$?
3. $\mathsf{pslf}_k(t)$ of a VM which provides $Q_1 = 4, Q_2 = 2$ budgets, on two CPUs, every period $P = 6$?

[Symmetric worst-case supply pattern (idea by Artem Burmyakov)]



How does it compare against $Q_1 = Q_2 = 3$ or against $Q_1 = 6$, $Q_2 = 0$?

Is even budget always worst?

If periods are not synchronized… ($P = 8$, $Q_1 + Q_2 = 8$)